

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Inventor : Hirokazu MATSUURA, et al.
Filed : Concurrently herewith
For : MULTIPROCESSOR SYSTEM
Serial No. : Concurrently herewith

November 24, 2003

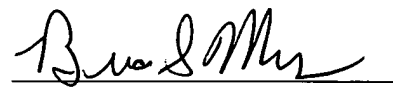
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

PRIORITY CLAIM AND
SUBMISSION OF PRIORITY DOCUMENT

S I R:

Applicant hereby claims priority under 35 USC 119 from **Japanese** patent application number **2002-345371** filed **November 28, 2002**, a certified copy of which is enclosed.

Respectfully submitted,



Brian S. Myers
Reg. No. 46,947

Katten Muchin Zavis Rosenman
575 Madison Avenue
New York, NY 10022-2585
(212) 940-8800
Docket No.: FUSA 20.757

日 本 国 特 許 庁
JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出 願 年 月 日 2 0 0 2 年 1 1 月 2 8 日
Date of Application:

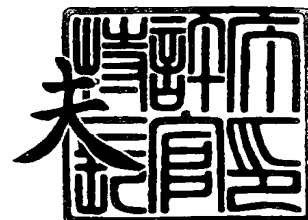
出 願 番 号 特 願 2 0 0 2 - 3 4 5 3 7 1
Application Number:
[ST. 10/C]: [J P 2 0 0 2 - 3 4 5 3 7 1]

出 願 人 富 士 通 株 式 会 社
Applicant(s):

2 0 0 3 年 7 月 1 8 日

特許庁長官
Commissioner,
Japan Patent Office

今 井 康 夫



【書類名】 特許願

【整理番号】 0251129

【提出日】 平成14年11月28日

【あて先】 特許庁長官殿

【国際特許分類】 G06F 15/16
G06F 12/08

【発明の名称】 マルチプロセッサシステム

【請求項の数】 5

【発明者】

【住所又は居所】 神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内

【氏名】 松浦 寛和

【発明者】

【住所又は居所】 神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内

【氏名】 村上 孝夫

【発明者】

【住所又は居所】 神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内

【氏名】 宇野 和也

【特許出願人】

【識別番号】 000005223

【氏名又は名称】 富士通株式会社

【代理人】

【識別番号】 100084711

【弁理士】

【氏名又は名称】 齋藤 千幹

【電話番号】 043-271-8176

【手数料の表示】

【予納台帳番号】 015222

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 9704946

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 マルチプロセッサシステム

【特許請求の範囲】

【請求項1】 共通メモリと多数のプロセッサとを共通バスを介して接続し、共通メモリの同一データ領域へのアクセスを1つのプロセッサのみに許可するマルチプロセッサシステムにおいて、

共通メモリにデータを記憶する多数のデータ領域と該データ領域毎の使用中表示制御情報を記憶する制御情報領域を設け、

各プロセッサに前記共通メモリと同等の記憶ユニットとアクセス制御部を設け、

アクセス権のないプロセッサのアクセス制御部は、共通バス上を流れるデータ及びアドレスを監視し、共通メモリへの書き込みデータ及び共通メモリからの読み取りデータを取り込んで自プロセッサ内の前記記憶ユニットに格納する、

ことを特徴とするマルチプロセッサシステム。

【請求項2】 各プロセッサの前記記憶ユニットと共通メモリのアドレス空間に同一のアドレスを割り振り、アクセス権のないプロセッサのアクセス制御部は、共通バス上のアドレスが指示する記憶ユニットの記憶域に共通バス上のデータを書込む、

ことを特徴とする請求項1記載のマルチプロセッサシステム。

【請求項3】 各プロセッサのアクセス制御部は、上位ユニットから前記共通メモリにおける所定データ領域へのアクセスが要求されたとき、前記記憶ユニットにおける該データ領域に応じた制御情報を読み取って、他プロセッサが使用中であるか判断し、使用中の場合には共通メモリにアクセスすることなく判断結果を上位装置に入力する、

ことを特徴とする請求項1記載のマルチプロセッサシステム。

【請求項4】 共通メモリと多数のプロセッサとを共通バスを介して接続し、共通メモリの同一データ領域へのアクセスを1つのプロセッサのみに許可するマルチプロセッサシステムにおいて、

共通メモリにデータを記憶する多数のデータ領域と該データ領域毎の使用中表示

示す制御情報を記憶する制御情報領域を設け、

各プロセッサに前記制御情報領域と同等の記憶ユニットとアクセス制御部を設け、

アクセス権のないプロセッサのアクセス制御部は、共通バス上を流れる制御情報及びアドレスを監視し、該制御情報を取り込んで自プロセッサ内の前記記憶ユニットに格納する、

ことを特徴とするマルチプロセッサシステム。

【請求項 5】 共通メモリと多数のプロセッサとを共通バスを介して接続し、共通メモリの同一データ領域へのアクセスを1つのプロセッサのみに許可するマルチプロセッサシステムにおいて、

共通メモリにデータを記憶する多数のデータ領域と該データ領域毎の使用中表示す制御情報を記憶する制御情報領域を設け、

各プロセッサに前記データ領域と同等の記憶ユニットと、アクセス制御部を設け、

アクセス権のないプロセッサのアクセス制御部は、共通バス上を流れるデータ及びアドレスを監視し、該データを取り込んで自プロセッサ内の前記記憶ユニットに格納する、

ことを特徴とするマルチプロセッサシステム。

【発明の詳細な説明】

【0 0 0 1】

【発明の属する技術分野】

本発明は、マルチプロセッサに係わり、特に、共有メモリを備えたマルチプロセッサシステムに関する。

【0 0 0 2】

【従来の技術】

図9は移動体通信システムの構成図であり、固定網1は市内電話交換機 1 a、1 b、中継交換機 1 c、関門交換機 1 d 等で構成されている。移動網2は、移動体用関門交換機 2 a、移動機に対してサービスを提供するための加入者クラスや位置登録情報を管理するホームロケーションレジスタ (HLR) 2 b、ホームロケーショ

ンレジスタ (HLR) を参照して無線制御を行う基地局制御装置 (RNC) 3a, 3b, を決定する移動体交換機 2 c 等で構成されている。

基地局制御装置 (RNC) 3a, 3b は、上位装置である移動体交換機 2 c やマルチメディア信号処理装置 (MPE) 4 と接続してコマンドを入出力する機能、コマンドに従った処理を行う機能、基地局 5 a, 5 b との間でデータの送受信を行う機能を備えている。各基地局 5a~5b は管理下のセル内に存在する多数のユーザ装置 (移動局) 6 a, 6 b と無線で通信する。固定網 1 には固定電話機 7 や PHS 用アダプタ 8 を介して PHS 基地局 9 が接続されている。

【 0 0 0 3 】

基地局制御装置 3a~3b はマルチプロセッサ構成になっている。図 10 は基地局制御装置のマルチプロセッサ構成であり、外部との送受信部は省略している。共通バスである G バス 10 にはバス割り当て制御を行うバスアービタ (BCONT) 11、複数のプロセッサ (CPU#1~CPU#n) 12₁~12_n、共通メモリカード (CM) 13 が接続されている。各プロセッサ CPU#1~CPU#n は同一の構成を備え、マイクロプロセッサユニット (MPU) 21 と、共通メモリカード (CM) 13 へのアクセスを制御するアクセス制御部 (LSI) 22 を有している。共通メモリカード (CM) 13 は RAM 構成の共通メモリ 23 と共通メモリコントローラ (CM コントローラ) 24 を有している。共通メモリ 23 は多数のデータ記憶領域 A~N を備えたデータ記憶部 GM と、それぞれの記憶領域 A~N が現在使用中であるか否か、並びに使用中のプロセッサ名を保存する排他制御データ記憶部 (セマフォレジスタ) semf を有している。使用中フラグが立っている記憶領域には他のプロセッサはアクセスすることができない。

【 0 0 0 4 】

バスアービタ 11 と各プロセッサ CPU#1~CPU#n 間は図 11 で示すようにそれぞれ信号線で接続されている。各プロセッサはバスが使用中でないと該信号線よりバス使用権要求信号 *BREQ_j をバスアービタに送出してバス使用権を要求する。バスアービタ 11 は所定のプロセッサにバス使用権を与える場合には、バス使用許可信号 *BGNT_j を該プロセッサに送出する。プロセッサはデータ転送中、バスビジー *BB を出力する必要があり、他のプロセッサはバスビジー *BB を参照してバス使用中であればバスを使用することが出来ない。

【0005】

所定のプロセッサ例えばプロセッサCPU#nがバス使用权を得て、データ記憶部GMのデータ記憶領域GM-Aにアクセスする場合、該データ記憶領域GM-Aが使用中でなければ対応するセマフォレジスタsemfの記憶領域aに使用中フラグ(排他制御フラグ)を立てる。しかる後、プロセッサCPU#nはデータ記憶領域GM-Aにアクセスし、データの読み出し、あるいはデータの書き込みを行い、データ記憶領域GM-Aにアクセスする必要が無くなればセマフォレジスタsemfの記憶領域aに記憶されている排他制御フラグをクリアする。これにより、他のプロセッサCPU#1~CPU#n-1はデータ記憶領域GM-Aにアクセス可能となる。

【0006】

ところで、従来のマルチプロセッサ処理での共通メモ리카ードCMへのアクセス方法だと、プロセッサCPU#nが使用中のデータ記憶領域GM-Aへアクセスしたい他のプロセッサCPU#1~CPU#n-1は、バス使用权を得る毎にセマフォレジスタsemfの内容を読み取ってデータ記憶領域GM-Aに応じた排他制御フラグが立っているか否かを調べる必要がある。排他制御フラグが立っていると2 μ sのタイミング後、同様に排他制御フラグが立っているか調べ、以後、フラグが解除されるまで無限に共通メモ리카ードCMへのセマフォレジスタsemfのリード命令を実行する。このため、図12に示すようにバスアクセス回数(バス使用回数)が非常に多くなるばかりでなく、このアクセス処理(無限リトライ)の実行回数の増大により、CPUの使用率の急上昇という結果になっていた。そのため、装置本来の能力が十分に発揮されないという結果になっていた。

【0007】

図13~図17は従来の共通メモ리카ードCMへのアクセス制御説明図であり、説明を簡単にするためにプロセッサはCPU#0, CPU#1の2台としている。セマフォレジスタ(排他制御データ記憶部)semfは、共通メモリにおけるデータ記憶領域A~Nのアクセス権を管理し、管理情報として、①アクセス権獲得中CPU番号と②獲得中フラグ(排他制御フラグ)を使用する。セマフォレジスタsemfが管理する共通メモリCMのアドレス範囲に関して、semf-aはデータ記憶領域GM-A、semf-bはデータ記憶領域GM-B、...、semf-nはデータ記憶領域GM-Nを管理するよう対応している

。

【0008】

所定のデータ領域に対するアクセス権を獲得するには、図13を参照すれば、以下の処理を行う。

①：CPU#0内のMPU 21からセマフォ獲得するためのリードアクセスが発生する。この時のターゲットは共通メモリカードCM 13の共通メモリ23のセマフォレジスタsemfにあるsemf-aである。

②：CPU#0内のアクセス制御部(LSI)は①のアクセス要求を受信し、共通バス10に対してリードアクセスを送出し、共通メモリカードCMからsemf-aの獲得結果の返送を待つ。

③：CMコントローラ24は共通バス上に送出された前記リードアクセスを受信し、そのアクセスに該当するsemf-aのデータを読み出す。

④：読み出したsemf-aのデータが未使状態であれば、今回のアクセス元のCCPU#0が使用中となるような管理情報を付加し、semf-aに対して書き戻す。これにより、semf-aはCPU#0の使用中状態となる。

⑤：ついで、CMコントローラ24はsemf-aに書き戻した管理情報を共通バス上に返送データとして送出する。返送データはCPU#0が獲得成功となっている。

⑥：CPU#0のアクセス制御部22は共通バス上に返送されたデータを受信し、MPU 21へ中継しセマフォ獲得処理が完了する。この時点でCPU#0はデータ領域GM-Aのアクセス権を獲得したことになる。

【0009】

以上により、semf-aに応じたデータ領域GM-Aのアクセス権が獲得されれば、CPU#0は図14に示すようにアクセスを開始する。

①：CPU#0が処理を開始し、その途中で共通メモリ23のデータ領域GM-Aからのデータの読み出し、あるいは書き込みが必要となれば、該データ領域GM-AへのアクセスがMPU 21から送出される。

②：アクセス制御部22は①のアクセスを受信し、共通バス10に対してリードアクセスを送出し、共通メモリカードCMのデータ領域GM-Aからの読み出しデータの返送を待つ。なお、例としてリードアクセスを記述する。

- ③：CMコントローラ24は共通バス上に送出されたリードアクセスを受信する。
- ④：CMコントローラ24は、そのリードアクセスに該当するデータ領域GM-Aのデータを読み出す。ライトの場合にはバス上のデータを受信し、GM-Aに書き込む。
- ⑤：CMコントローラ24は、読み出したデータを共通バス上に返送データとして送出する。
- ⑥：CPU#0のアクセス制御部22は共通バス上に返送されたデータを受信し、MPU 21へ中継する。

以降、CPU#0の処理が終了するまで、①～⑥が繰り返されることになり、ライトアクセスが発生する毎に、データ領域GM-Aのデータが更新される。

【0 0 1 0】

CPU#0のデータ領域GM-Aへのアクセスが不要になれば図15に示すように終了処理が行われる。

①：CPU#0の処理が終了すれば、MPU21はデータ領域GM-Aのアクセス権を開放するために、開放アクセスをsemf-aに向けて送出する。すなわち、MPU21はsemf-aへのデータ0のライトアクセスを行う。

②：このライトアクセスは、アクセス制御部22が一旦受信し、突き放し処理により、MPU 21に向けて終了報告をする。この時点で、MPU側の処理は完了するが、後処理をアクセス制御部22が実施する。

③：ライトアクセスを受信したアクセス制御部22はsemf-a向けのライトアクセスを共通バス上に送出する。

④：CMコントローラ24は共通バス上に送出されたアクセス（ライトアドレスとデータ）を受信する。

⑤：CMコントローラ24は受信したライトデータ（=0）をセマフォレジスタsemfのsemf-aに書き込み、該semf-aを未使用状態に戻す。これにより、以降、他CPUからもデータ領域GM-Nへアクセス可能状態になる。

これら一連の処理により、CPU#0が共通メモリカードCMのデータ領域GM-Aを更新できることになる。

【0 0 1 1】

一方、図13の動作でsemf-aに応じたデータ領域GM-AのアクセスがCPU#0により

獲得されている状態において、CPU#1よりセマフォ獲得するためのリードアクセスが発生すると、図16に示す処理が行われる

①: CPU#1内のMPU 21はセマフォ獲得するためのリードアクセスを出力する。この時のターゲットは共通メモリカードCMの共通メモリ23上にあるセマフォレジスタのsemf-aである。

②: CPU#1内のアクセス制御部22は①のアクセスを受信すれば、共通バス10に対してリードアクセスを送出し、共通メモリカードCMからsemf-aの獲得結果の返送を待つ。

③: CMコントローラ24は共通バス上に送出されたリードアクセスを受信し、そのアクセスに該当するセマフォレジスタのsemf-aのデータを読み出す。

④: ついで、CMコントローラ24は読み出したsemf-aのデータが使用中状態であるか判別する。使用中であるため、アクセス元のCPU#1のセマフォ獲得は失敗となる。なお、管理情報はCPU#0が使用中のため、特に書き戻し処理は発生しない。semf-aはCPU#0使用中状態のままである。

⑤: CMコントローラ24はCPU#1に対して、semf-a獲得失敗の情報を共通バス上に返送データとして送出する。返送データはCPU#1がsemf-a獲得失敗となっている。

⑥: CPU#1のアクセス制御部22は共通バス上に返送されたデータを受信し、MPU 21へ中継する。

この時点でCPU#1はsemf-aに応じたデータ領域MG-Aへのアクセス権の獲得を失敗したことになり、以後、semf-a（データ領域MG-Aへのアクセス権）が獲得できるまでリトライアクセスを繰り返すことになる。

【0012】

semf-aに応じたデータ領域MG-Aへのアクセス権がすでにCPU#0により獲得されており、データ領域GM-Aに対してCPU#0がアクセス中である場合、CPU#1がsemf-aへアクセス権獲得のリトライアクセスを繰り返したとき、図17に従って以下の処理が行われる。CPU#0のアクセス処理は図14の①～⑥と同じである。かかる状態において、CPU#0の処理の合間にCPU#1のセマフォ獲得リトライアクセスが発生すると、⑦以降の処理(図16の①～⑥に相当)が行われる。このため、共通バ

ス10を圧迫させることになる。

その結果、CPU#0から出力されるアクセスが遅延し、CPU#0が獲得しているsemf-aの開放処理が遅れる。また、その期間中もCPU#1からのセマフォ獲得リトライアクセスが発生してしまうことにより、CPU#0、CPU#1両方の処理が遅延することになる。

【0013】

共通メモリを備えたマルチプロセッサシステムとして、特開平6-274415号公報に開示されたシステムがある。このマルチプロセッサシステムでは、各プロセッサにキャッシュメモリを設け、プロセッサが共通メモリにアクセスしたとき、アクセスの種別(読み/書きの別)及び読み出しあるいは書込んだデータをキャッシュメモリに保存しておき、他装置が共通メモリにデータを書込んだとき、該キャッシュメモリのデータを無効にするものである。このようにすることにより、バス状態を監視することなく、共有メモリとキャッシュメモリの整合性を保持でき、しかも、キャッシュメモリに記憶されているデータにヒットすれば共通メモリをアクセスする必要が無い。

【特許文献1】 特開平6-274415号公報

【0014】

【発明が解決しようとする課題】

図10~図17に示した従来のマルチプロセッサシステムでは、バスアクセス回数(バス使用回数)が非常に多くなり、使用中プロセッサから出力されるアクセスが遅延し、また、獲得しているsemf-aの開放処理も遅れる。更に、アクセスリトライ処理の実行回数の増大により、CPU使用率の急上昇を招き、スループットを低下させる問題がある。

特開平6-274415号公報のマルチプロセッサシステムは、キャッシュメモリを設ける点で有用性を発生するが、共通メモリへのアクセスが競合する場合の解決方法を与えない。

以上から本発明の目的は、各プロセッサからの共通メモリへのアクセスが競合しても、共通メモリへのアクセス回数を削減して、プロセッサの能力を十分に発揮できるようにすることである。

【0015】

【課題を解決するための手段】

本発明は、共通メモリと多数のプロセッサとを共通バスを介して接続し、共通メモリの同一データ領域へのアクセスを1つのプロセッサのみに許可するマルチプロセッサシステムである。

このマルチプロセッサシステムにおいて、①共通メモリにデータを記憶する多数のデータ領域と該データ領域毎の使用中表示する制御情報を記憶する制御情報領域とを設け、②各プロセッサに前記共通メモリと同等の記憶ユニットとアクセス制御部を設け、③アクセス権のないプロセッサのアクセス制御部は、共通バス上を流れるデータ及びアドレスを監視し、共通メモリへの書き込みデータ及び共通メモリからの読み取りデータを取り込んで自プロセッサ内の前記記憶ユニットに格納する。即ち、各プロセッサの前記記憶ユニットと共通メモリのアドレス空間に同一のアドレスを割り振り、アクセス権のないプロセッサのアクセス制御部は、共通バス上のアドレスが指示する自分の記憶ユニットの記憶域に共通バス上のデータを書込む。

【0016】

以上のように構成することにより、各プロセッサの記憶ユニットと共通メモリの内容を同一に出来る。この結果、データ領域のアクセス権を獲得していないプロセッサ装置は、自装置内の記憶ユニットを参照するだけで他のプロセッサ装置が該データ領域を使用しているかわかる。このため、従来のように一々共通バスを介して共通メモリに問い合わせる必要がなく、プロセッサのスループットを向上することができ、しかも、短時間で結果を入手することが出来る。

又、共通バスがビジーとなる状態を少なく出来るため、アクセス権を獲得しているプロセッサは共通メモリから高速にデータの読み出しやデータの書き込みが可能になる。

又、本発明のマルチプロセッサシステムにおいて、所定データ領域のアクセス権を獲得しているプロセッサは、自装置内の記憶ユニットにヒットすれば共通メモリにアクセスする必要がなく、高速に該記憶ユニットから該データを読み取って処理することが出来、プロセッサのスループットを向上することができる。

【0017】

【発明の実施の形態】

(A) 第1実施例

(a) マルチプロセッサシステムの構成

図1は第1実施例のマルチプロセッサシステムの構成図である。

共通バスであるGバス50にはバス割り当て制御を行うバスアービタ (BCONT) 51、複数のプロセッサ (CPU#1~CPU#n) 52₁~52_n、共通メモリカード (CM) 53が接続されている。各プロセッサCPU#1~CPU#nは同一の構成を備え、マイクロプロセッサユニット (MPU) 61と、共通メモリカード (CM) 53へのアクセスを制御するアクセス制御部 (専用LSI) 62と、キャッシュメモリ63を有している。

【0018】

共通メモリカード (CM) 53はRAM構成の共通メモリ71と共通メモリコントローラ (CMコントローラ) 72を有している。共通メモリ71は多数のデータ記憶領域A~Nを備えたデータ記憶部GMと、それぞれのデータ記憶領域が現在使用中であるか否か、並びに使用中のプロセッサ名を保存するセマフォレジスタ (排他制御データ記憶部) semfを有している。データ領域A~Nの使用状況を記憶するセマフォレジスタ部分はsemf-a~semf-nである。使用中フラグが立っているsemf-jに応じたデータ記憶領域Jには他のプロセッサはアクセスすることができない。

各プロセッサのキャッシュメモリ63は、共通メモリカードCMの共通メモリ71と同一サイズで同一の情報を記憶できるようになっている。すなわち、キャッシュメモリ63は共通メモリ71と同様に、①データ記憶領域A~Nを備えたデータ記憶部GM (GMキャッシュ) と、②各データ記憶領域が現在使用中であるか否か、並びに使用中のプロセッサ名を保存するセマフォレジスタsemf (セマフォキャッシュ) を有している。

【0019】

バスアービタ11と各プロセッサCPU#1~CPU#n間は従来例 (図11) と同様にそれぞれ信号線で接続されている。各プロセッサはバスが使用中でないと該信号線よりバス使用权要求信号*BREQ_jをバスアービタに送出してバス使用权を要求する。バスアービタ11は所定のプロセッサにバス使用权を与える場合には、バス使用使

用許可信号*BGNTjを該プロセッサに送出する。プロセッサはデータ転送中、バスビジー*BBを出力する必要がある、他のプロセッサはバスビジー*BBを参照してバス使用中であればバスを使用することが出来ない。

【0020】

(b)本発明の概略動作

本発明の概略動作を説明する。例えばCPU#nからの要求により、CMコントローラ72が該CPU#nにデータ記憶領域aのアクセス権を与えたとする。かかる場合、CMコントローラ72は、共通メモ리카ード53のセマフォレジスタsemfのsemf-aに管理情報(使用中フラグ、使用中プロセッサ名)を書込むと共に、該管理情報をCPU#nのセマフォキャッシュ63bのsemf-aに書込むライトアクセスを共通バス50上に出送する。CPU#nのアクセス制御部(専用LSI)62は該管理情報を取り込んでセマフォキャッシュ63bのsemf-aに書込む。このとき、他のCPU#1～CPU#n-1は、共通バス上にアドレス/データが流れているか監視しており、該アドレス/データを盗み取りし、あたかも自身への書き込みデータであるかのように取り込んで自身のキャッシュメモリ63のセマフォレジスタ63bのsemf-aに格納する。セマフォsemf-aを解放する場合も同様である。従って、全CPU#1～CPU#nおよび共通メモ리카ードにおけるセマフォレジスタの内容はすべて同じになる。

【0021】

CPU#nが、共通メモ리카ード53のデータ記憶領域GM-Aからデータを読み出すとき以下の制御が行われる。すなわち、CPU#nが共通メモ리카ード53にリードアクセスすると、共通メモ리카ード53は共通メモリ71よりデータを読み出して共通バス50に送出する。CPU#nは該データを共通バス50を介して受信して内蔵のMPU 61に入力すると共にキャッシュメモリ63のGMキャッシュ63aのデータ領域GM-Aに格納する。この時、他のCPU#1～CPU#n-1は、共通バス上にアドレス/データが流れているか監視しており、該アドレス/データを盗み取りし、自身のキャッシュメモリ63のGMキャッシュ63aのデータ領域GM-Aに格納する。

【0022】

又、CPU#nが、共通メモ리카ード53のデータ記憶領域GM-Aにデータを書込むとき以下の制御が行われる。すなわち、CPU#nは共通メモ리카ード53に共通バス50を介

してライトアクセスすると共に、自身のキャッシュメモリ63のGMキャッシュ63aのデータ領域GM-Aにライトアクセスにかかわるデータを書込む。また、共通メモリカード53のCMコントローラ72は該ライトアクセスによりデータを共通メモリ71のデータ領域GM-Aに書込む。このとき、他のCPU#1～CPU#n-1は、共通バス上にアドレス/データが流れているか監視しており、該アドレス/データを盗み取りし、自身のキャッシュメモリ63のGMキャッシュ63aのデータ領域GM-Aに格納する。

以上により、全CPU#1～CPU#nおよび共通メモリカードにおけるデータ記憶部GMの内容が同じになる。

【0023】

従って、各CPU#1～CPU#nは、自身のセマフォレジスタsemfを参照すればデータ格納領域A～Nの使用中の有無を識別でき、従来技術のように共通バスを介して一々共通メモリカードのセマフォレジスタを参照する必要が無くなり、アクセス回数を減少することができる。

又、データ格納領域GM-jのアクセス権を得たCPUはヒットする場合には共通メモリカードにアクセスしなくても、自身のキャッシュメモリのデータ記憶部からデータを読み出すことができ、共通メモリカードへのアクセス回数を減らすことができ、各プロセッサの能力を十分に発揮できるようになる。

【0024】

(c) 本発明の詳細動作

図2は、CPU#0が所定のデータ領域(データ領域N)に対するアクセス権を獲得する処理の説明図である。なお、各CPU#0～CPU#1のキャッシュメモリ63及び共通メモリ71には同一アドレスが割り振られている。

①: CPU#0内のMPU 61はセマフォ獲得するためのリードアクセスを発生する。この時のターゲットは共通メモリカードCM上にあるsemf-nである。

②: アクセス制御部(専用LSI)62は、①のアクセスを受信すると、CPU#0内のセマフォキャッシュ63bから該当アドレスの状態を読み出す。読み出した結果、他CPUが使用中でなければ、すなわち、空き状態であれば、専用LSI 62は共通バス50に対してリードアクセスを送出し、共通メモリカードCMからのsemf-nの獲得結果の返送を待つ。

③：CMコントローラ72は共通バス上に送出されたリードアクセスを受信し、そのアクセスに該当するsemf-nのデータを読み出す。

④：読み出したsemf-nのデータが未使状態であれば、今回のアクセス元のCPU#0が使用中となるような管理情報を作成してsemf-nに書き戻す。これにより、semf-nはCPU#0使用中状態となる。尚、管理情報に使用中フラグデータ及び使用プロセッサ名を含める。

⑤：CMコントローラ72は、semf-nに書き戻した管理情報及びキャッシュアドレスを共通バス50上に返送データとして送出する。返送データはCPU#0が獲得成功となっている。

⑥：CPU#0の専用LSI 62は共通バス上に返送されたデータを受信し、MPU61へ中継する。この時点でCPU#0はデータ領域GM-Aのアクセス権を獲得したことになる。

⑦：また、専用LSI 62はこの時のデータ(管理情報)を、アドレスが指定するセマフォキャッシュ63bのsemf-nにも書き込む。これにより、セマフォキャッシュはCPU#0使用中状態を示す。この時点で、CPU#0内のsemf-nと共通メモリカードCM内semf-nの内容が一致することになる。

⑧：一方、CPU#0以外のCPU#1は、⑤で共通バス上に送出されているアドレス、データ等を監視しており、その時のアクセスを盗み取りし、CPU#1内の専用LSI 62も受信することになる。

⑨：CPU#1内の専用LSI 62は、取り込んだデータを、アドレスが指定するセマフォキャッシュ63bのsemf-nに書き込む。この時点で、CPU#0、CPU#1および共通メモリカードCM内のsemf-nの内容がすべて一致することになる。

【0 0 2 5】

図3は、semf-n(データ領域GM-Nのアクセス権)がCPU#0により獲得されており、CPU#0、CPU#1及び共通メモリカードCM内のsemf-nの内容が一致している状態において、CPU#0がデータ領域GM-Nにアクセスする処理説明図である。

①：CPU#0のMPU 61が処理を開始し、その途中でデータ領域GM-Nからデータを読み出し、あるいはデータ領域GM-Nへデータを書き込む必要が発生した場合、MPU 61はデータ領域GM-Nへのアクセスを出力する。

②：CPU#0の専用LSI 62は、①のアクセスを受信し、CPU#0内のGMキャッシュ63aから該当アドレスの状態を読み出す。なお、図にはアドレスの状態を記憶する領域は示していないが、該領域には各アドレスのデータの有効、無効を表示するステータス情報が保持されている。

読み出した結果、該当アドレスのデータが無効であれば、専用LSI 62は共通バス50に対してリードアクセスを送出し、共通メモ리카ードCMのデータ領域GM-Nからのデータ返送を待つ。

なお、GMキャッシュ63aから読み出したステータ情報がデータの有効を示していれば、GMキャッシュ63aのデータ領域GM-Nからデータを読み出してMPU 61に入力して処理は終了する。

③：データが無効で、専用LSI 62が共通バス50に対してリードアクセスを送出すれば、CMコントローラ72は共通バス上に送出されたリードアクセスを受信する。

④：ついで、CMコントローラ72は該当するデータ領域GM-Nのデータを読み出す。

⑤：CMコントローラ72は読み出したデータ領域GM-Nのデータとデータ領域GM-Nのアドレスを共通バス上に返送データとして送出する。

⑥：CPU#0内の専用LSI 62は共通バス上に返送されたデータを受信し、MPU 61へ中継する。

⑦：また、CPU#0内の専用LSI 62はこの時のデータをアドレスが指定するGMキャッシュ63aのデータ領域GM-Nにも書き込む。この時点で、CPU#0内のデータ領域GM-Nと共通メモ리카ードCM内のデータ領域GM-Nの内容が一致することになる。

⑧：以上と並行して、CPU#0以外のCPU#1は、⑤で共通バス上に送出されているアドレス、データ等を監視しており、その時のアクセスを盗み取りし、CPU#1内の専用LSI 62も受信することになる。

⑨：CPU#1内の専用LSI 62は、共通バスに送出されたデータ領域GM-Nのデータを取り込み、アドレスが指定するGMキャッシュ63a内のデータ領域GM-Nに書き込む。この時点で、CPU#0、CPU#1および共通メモ리카ードCM内のデータ領域GM-Nの内容がすべて一致することになる。

以降、CPU#0 の処理が終了するまで、①～⑨が繰り返される。また、ライトアクセスが発生する毎に、データ領域GM-Nのデータが更新される。

以降、同様に、各CPU内の専用LSI 62により共通バス上のアクセスが監視されることで、CPU#0, CPU#1 および共通メモリカードCM内のデータ領域GM-Nの内容がすべて一致する。

【0026】

図4はデータ記憶部GMへの書き込みの処理である。

①: CPU#0 が処理を開始し、その途中でデータ領域GM-Nへの書き込みが必要になるとMPU 61は該データ領域GM-Nへのライトアクセスを出力する。

②: CPU#0 の専用LSI 62はそのアクセスを受信した時点で、MPU 61に対して正常応答しMPU間の通信処理を完了させる。ただし、実際には、共通メモリカードCM内のデータ領域GM-Nに対する書き込み処理が完了していないので、専用LSI 62は後処理を継続する。

③: 専用LSI 62は受信したデータ領域GM-Nに書込むデータとデータ領域GM-Nのアドレスを共通バス上に送出する。

④: ③と同時に専用LSI 62は、CPU#0 内のGMキャッシュ63aから該当アドレスの状態を読み出す。書き込みの場合は、該当アドレスのデータの無効/有効に関係なくCPU#0 のGMキャッシュ63a内のデータ領域GM-Nにライトデータを書き込む。

⑤: CMコントローラ72は共通バス上に送出されたデータを受信する。

⑥: ついで、CMコントローラ72は受信した該当アドレスに対するデータ領域GM-Nに対してライトデータを書き込む。

⑦: CPU#0 以外のCPU#1 は、③で共通バス上に送出されているアドレス、データ等を監視しており、その時のアクセスを盗み取りし、CPU#1 内の専用LSI 62がこれらアドレス、データを取り込む。

⑧: CPU#1 内の専用LSI 62は、盗み取りしたデータをアドレスが指示するGMキャッシュ63aのデータ領域GM-Nに書き込む。この時点で、CPU#0, CPU#1 および共通メモリカードCM内のデータ領域GM-Nの内容が一致することになる。

以後、CPU#0 からのライト処理が発生するたびに、①～⑧が繰り返されデータ

領域GM-Nのデータが更新される。

以降、各CPU内の専用LSIにより共通バス上のアクセスが監視されることで、各GMキャッシュ63a及び共通メモ리카ードのデータ記憶部GMの内容が一致することになる。

【0027】

図5は共通メモ리카ードのデータ記憶部GMからの読み出しが、CPU#0内のGMキャッシュ63aにヒットした場合の処理である。

①：CPU#0が処理を開始し、その途中でデータ領域GM-Nからのデータ読み出しが必要になると、MPU 61は該データ領域GM-Nへのリードアクセスを出力する。

②：CPU#0の専用LSI 62はそのアクセスを受信した時点で、CPU#0内のGMキャッシュ63aから該当アドレスの状態を読み出す。

③：専用LSI 62は読み出したデータが有効な場合には、そのデータをMPU 21に対して返送し、リードアクセスを完了する。この場合には、共通メモ리카ードCMへのリードアクセスは不要なため共通バスへのアクセスもキャンセルされる。

【0028】

図6はCPU#0がセマフォレジスタ63bのsemf-nに応じたデータ領域GM-Nを使用中において、CPU#1によるsemf-nのセマフォ獲得アクセスが発生した場合の処理である。

①：CPU#1のMPU 61はsemf-nに対するセマフォ獲得アクセスを出力する。なお、この時、semf-nはすでにCPU#0によって使用中状態である。

②：CPU#1の専用LSI 62は、受信したセマフォ獲得アクセスに該当するセマフォキャッシュ63bのアドレスからsemf-nを読み出し、CPU#0が使用中状態であることを認識する。

③：CPU#1の専用LSI 62はMPU 61に対してセマフォ獲得アクセスが失敗したデータを返送する。この時、CPU#1の専用LSI 62は共通メモ리카ードCMへのセマフォ獲得アクセスが不要となるため、共通バスへのアクセスをキャンセルする。

以後、CPU#1のMPU 61は、CPU#0による使用が終了してセマフォ獲得が成功するまで①から③の動作を繰り返す。この間、セマフォ獲得失敗時のリトライアクセスは共通バス上に送出されない。

【0029】

以上、第1実施例によれば、所定のデータ領域のアクセス権を獲得していないプロセッサ装置は装置内情報を参照するだけで他のプロセッサ装置が該データ領域を使用しているかわかるため、従来のように一々共通バスを介して共通メモリカードに問い合わせる必要がなく、プロセッサのスループットを向上することができ、しかも、短時間で結果を入手することが出来る。

又、第1実施例によれば、所定のデータ領域のアクセス権を獲得しているプロセッサは自装置内のキャッシュにヒットすれば共通メモリカードにアクセスする必要がなく高速にデータを読み取って処理することが出来、プロセッサのスループットを向上することができる。又、アクセス権のないプロセッサにデータの書き込みを禁止させ、データの読み取りを可能とすれば、他のプロセッサも自装置内のキャッシュにヒットすれば共通メモリカードにアクセスする必要がなく高速にデータを読み取って処理することが出来、プロセッサのスループットを向上することが可能となる。

又、共通バスがビジーとなる状態を少なくできるため、アクセス権を獲得しているプロセッサは共通メモリから高速にデータの読み出しやデータの書き込みが可能になる。

【0030】

(B) 第2実施例

図7は本発明の第2実施例の構成図であり、図1の第1実施例と同一部分には同一符号を付している。異なる点は、第2実施例においてキャッシュメモリ63をセマフォキャッシュ63bのみ設け、GMキャッシュ63aを省略した点である。

第2実施例では、セマフォキャッシュ63bに書込む管理情報（使用中フラグデータ及び使用プロセッサ名を含む）が共通バス50上を流れたとき（図2参照）、アクセス権のない他のプロセッサは、該データを盗み取りして自分のセマフォキャッシュ63bに保存する。

第2実施例によれば、第1実施例の図6と同様に、データ領域のアクセス権を獲得していないプロセッサ装置は装置内情報を参照するだけで他のプロセッサ装置が該データ領域を使用しているかわかる。このため、従来のように一々共通バスを

介して共通メモリに問い合わせる必要がなく、プロセッサのスループットを向上することができ、しかも、短時間で結果を入手することが出来る。

【0 0 3 1】

(C) 第3実施例

図8は本発明の第3実施例の構成図であり、図1の第1実施例と同一部分には同一符号を付している。異なる点は、第3実施例においてキャッシュメモリ63をGMキャッシュ63aのみ設け、セマフォキャッシュ63bを省略した点である。

第3実施例では、GMキャッシュ63aに書込むデータが共通バス50上を流れたとき(図3、図4参照)、アクセス権のない他のプロセッサは、該データを盗み取りして自分のGMキャッシュ63aに保存する。

第3実施例によれば、データ領域のアクセス権を獲得しているプロセッサは自装置内のキャッシュにヒットすれば共通メモリにアクセスする必要がなく高速にデータを読み取って処理することが出来る、プロセッサのスループットを向上することができる。

又、アクセス権のないプロセッサにデータの書き込みを禁止させ、データの読み取りを可能にすれば、他のプロセッサも自装置内のGMキャッシュ63aにヒットすれば共通メモリにアクセスする必要がなく、高速にデータを読み取って処理がすることが出来る、プロセッサのスループットを向上することができる。

【0 0 3 2】

(D) 変形例

図1の第1実施例では共通メモリカード53を残したが、初期時に、各プロセッサの記憶ユニット63の内容を同一にし、かつ、記憶内容をすべて有効にしておけば、この共通メモリカード53を削除することもできる。この場合、アクセス権を有するプロセッサCPU#nが共通メモリカード(実在しない)へデータを書き込む場合には、自分の記憶ユニット63に該データを書込むと共に、共通バス50にデータとアドレスを送出する。他のプロセッサCPU#1～CPU#n-1は共通バス上のデータとアドレスを取り込んで自分の記憶ユニット63に書込む。

【0 0 3 3】

(付記1) 共通メモリと多数のプロセッサとを共通バスを介して接続し

、共通メモリの同一データ領域へのアクセスを1つのプロセッサのみに許可するマルチプロセッサシステムにおいて、

共通メモリにデータを記憶する多数のデータ領域と該データ領域毎の使用中表示制御情報を記憶する制御情報領域を設け、

各プロセッサに前記共通メモリと同等の記憶ユニットとアクセス制御部を設け、

アクセス権のないプロセッサのアクセス制御部は、共通バス上を流れるデータ及びアドレスを監視し、共通メモリへの書き込みデータ及び共通メモリからの読み取りデータを取り込んで自プロセッサ内の前記記憶ユニットに格納する、

ことを特徴とするマルチプロセッサシステム。

(付記2) 各プロセッサの前記記憶ユニットと共通メモリのアドレス空間に同一のアドレスを割り振り、アクセス権のないプロセッサのアクセス制御部は、共通バス上のアドレスが指示する記憶ユニットの記憶域に共通バス上のデータを書込む、

ことを特徴とする付記1記載のマルチプロセッサシステム。

(付記3) 各プロセッサのアクセス制御部は、上位ユニットから前記共通メモリにおける所定データ領域へのアクセスが要求されたとき、前記記憶ユニットにおける該データ領域に応じた制御情報を読み取って、他プロセッサが使用中であるか判断し、使用中の場合には共通メモリにアクセスすることなく判断結果を上位装置に入力する、

ことを特徴とする付記1記載のマルチプロセッサシステム。

(付記4) アクセス権のあるプロセッサのアクセス制御部は、上位ユニットから共通メモリにおける所定データ領域からのデータ読み出しが指示されたとき、前記記憶ユニットにおける該データ領域が有効であれば、このデータ領域からデータを読み取って上位装置に入力する、

ことを特徴とする付記1記載のマルチプロセッサシステム。

(付記5) アクセス権のあるプロセッサのアクセス制御部は、上位ユニットから共通メモリにおける所定データ領域へのデータの書き込みが指示されたとき、前記記憶ユニットのデータ領域にデータを書込むと共に、該データ及び該

データ領域に応じたアドレスを共通バスに送出する、

ことを特徴とする付記2記載のマルチプロセッサシステム。

(付記6) 共通メモリと多数のプロセッサとを共通バスを介して接続し、共通メモリの同一データ領域へのアクセスを1つのプロセッサのみに許可するマルチプロセッサシステムにおいて、

共通メモリにデータを記憶する多数のデータ領域と該データ領域毎の使用中表示す制御情報を記憶する制御情報領域を設け、

各プロセッサに前記制御情報領域と同等の記憶ユニットとアクセス制御部を設け、

アクセス権のないプロセッサのアクセス制御部は、共通バス上を流れる制御情報及びアドレスを監視し、該制御情報を取り込んで自プロセッサ内の前記記憶ユニットに格納する、

ことを特徴とするマルチプロセッサシステム。

(付記7) 各プロセッサの前記記憶ユニットと、共通メモリの制御情報領域とに同一のアドレスを割り振り、アクセス権のないプロセッサのアクセス制御部は、共通バス上のアドレスが指示する記憶ユニットの記憶域に共通バス上の制御情報を取り込んで書込む、

ことを特徴とする付記6記載のマルチプロセッサシステム。

(付記8) 各プロセッサのアクセス制御部は、上位ユニットから前記共通メモリにおける所定データ領域へのアクセスが要求されたとき、前記記憶ユニットにおける該データ領域に応じた制御情報を読み取って、他プロセッサが使用中であるか判断し、使用中の場合には共通メモリにアクセスすることなく判断結果を上位装置に入力する、

ことを特徴とする付記6記載のマルチプロセッサシステム。

(付記9) 共通メモリと多数のプロセッサとを共通バスを介して接続し、共通メモリの同一データ領域へのアクセスを1つのプロセッサのみに許可するマルチプロセッサシステムにおいて、

共通メモリにデータを記憶する多数のデータ領域と該データ領域毎の使用中表示す制御情報を記憶する制御情報領域を設け、

各プロセッサに前記データ領域と同等の記憶ユニットと、アクセス制御部を設け、

アクセス権のないプロセッサのアクセス制御部は、共通バス上を流れるデータ及びアドレスを監視し、該データを取り込んで自プロセッサ内の前記記憶ユニットに格納する、

ことを特徴とするマルチプロセッサシステム。

【0034】

【発明の効果】

以上本発明によれば、所定のデータ領域のアクセス権を獲得していないプロセッサ装置は装置内情報を参照するだけで他のプロセッサ装置が該データ領域を使用しているかわかる。このため、従来のように一々共通バスを介して共通メモリに問い合わせる必要がなく、プロセッサのスループットを向上することができ、しかも、短時間で結果を入手することが出来る。

又、本発明によれば、所定のデータ領域のアクセス権を獲得しているプロセッサは自装置内のキャッシュにヒットすれば共通メモリにアクセスする必要がなく高速にデータを読み取って処理することが出来、プロセッサのスループットを向上することができる。

又、本発明によれば、アクセス権のないプロセッサにデータの書き込みのみを禁止させ、データの読み取りを可能とすれば、他のプロセッサも自装置内のキャッシュにヒットすれば共通メモリにアクセスする必要がなく高速にデータを読み取って処理することが出来、プロセッサのスループットを向上することが可能となる。

又、本発明によれば、共通バスがビジーとなる状態を少なく出来るため、アクセス権を獲得しているプロセッサは共通メモリから高速にデータの読み出しやデータの書き込みが可能になる。

【図面の簡単な説明】

【図1】

第1実施例のマルチプロセッサシステムの構成図である。

【図2】

CPU#0が所定のデータ領域(データ領域GM-N)に対するアクセス権を獲得する処理の説明図である。

【図 3】

データ領域GM-Nのアクセス権がCPU# 0 により獲得されている状態において、該CPU# 0 がデータ領域GM-Nにアクセスする処理説明図である。

【図 4】

データ記憶部GMへの書き込みの処理説明図である。

【図 5】

共通メモ리카ードのデータ記憶部GMからの読み出しが、CPU# 0 内GMキャッシュにヒットした場合の処理説明図である。

【図 6】

CPU# 0 がsemf-n に応じたデータ領域GM-Nを使用中において、CPU# 1 によるsemf-nのセマフォ獲得アクセスが発生した場合の処理説明図である。

【図 7】

本発明の第2実施例の構成図である。

【図 8】

本発明の第3実施例の構成図である。

【図 9】

移動体通信システムの構成図である。

【図 1 0】

従来の基地局制御装置のマルチプロセッサ構成図である。

【図 1 1】

バスアービタと各プロセッサ間の接続図である。

【図 1 2】

従来のバスアクセス回数(バス使用回数)の状況説明図である。

【図 1 3】

従来の共通メモ리카ードCMへの第1のアクセス制御説明図である。

【図 1 4】

従来の共通メモ리카ードCMへの第2のアクセス制御説明図である。

【図 1 5】

従来の共通メモリカードCMへの第3のアクセス制御説明図である。

【図 1 6】

従来の共通メモリカードCMへの第4のアクセス制御説明図である。

【図 1 7】

従来の共通メモリカードCMへの第5のアクセス制御説明図である。

【符号の説明】

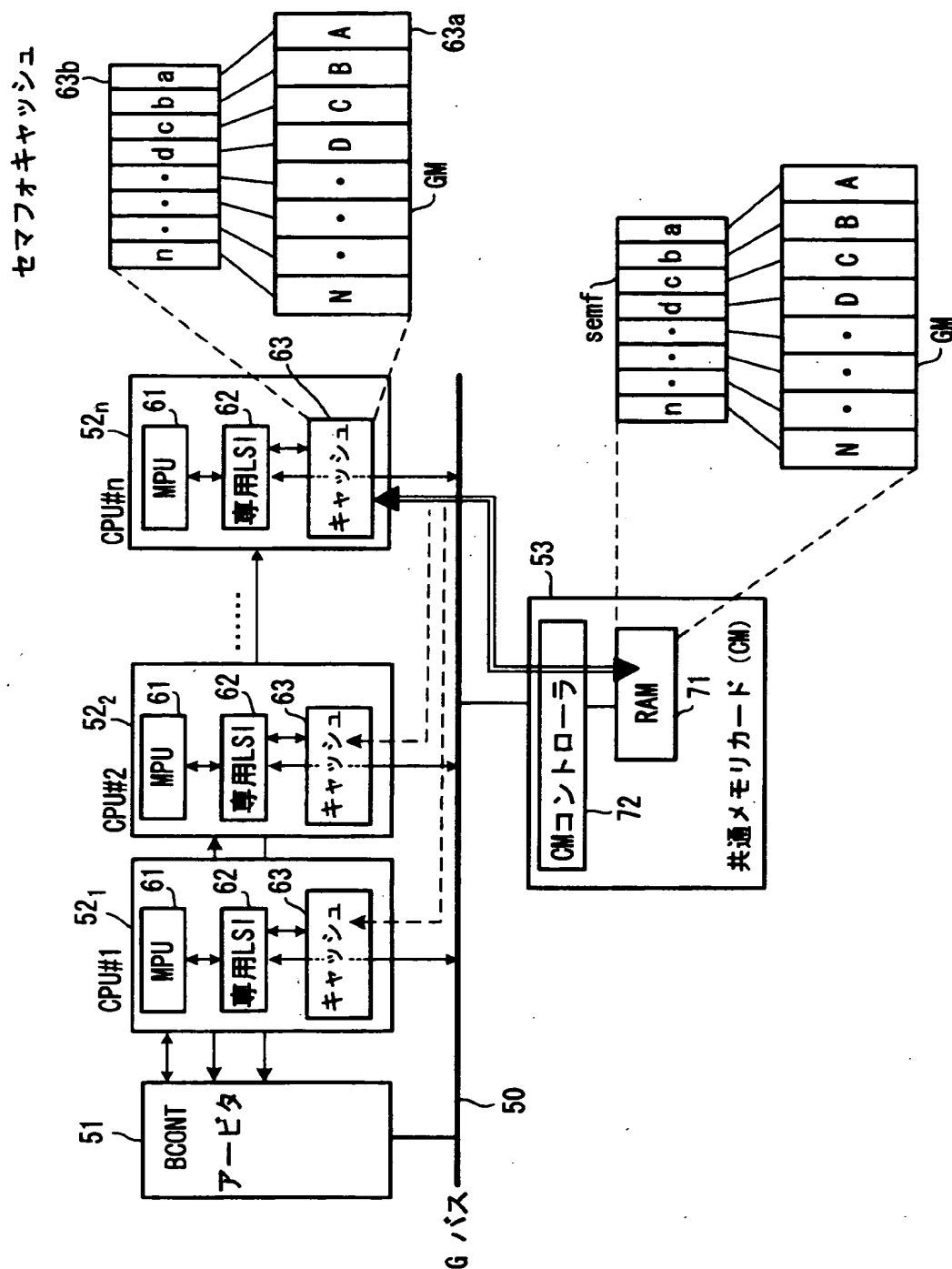
- 5 0 共通バスであるGバス
- 5 1 バスアービタ (BCONT)
- 5 2₁ ~ 5 2_n プロセッサ (CPU#1 ~ CPU#n)
- 5 3 共通メモリカード (CM)
- 6 1 マイクロプロセッサユニット (MPU)
- 6 2 アクセス制御部 (専用LSI)
- 6 3 キャッシュメモリ
- 7 1 共通メモリ
- 7 2 共通メモリコントローラ (CMコントローラ)

【書類名】

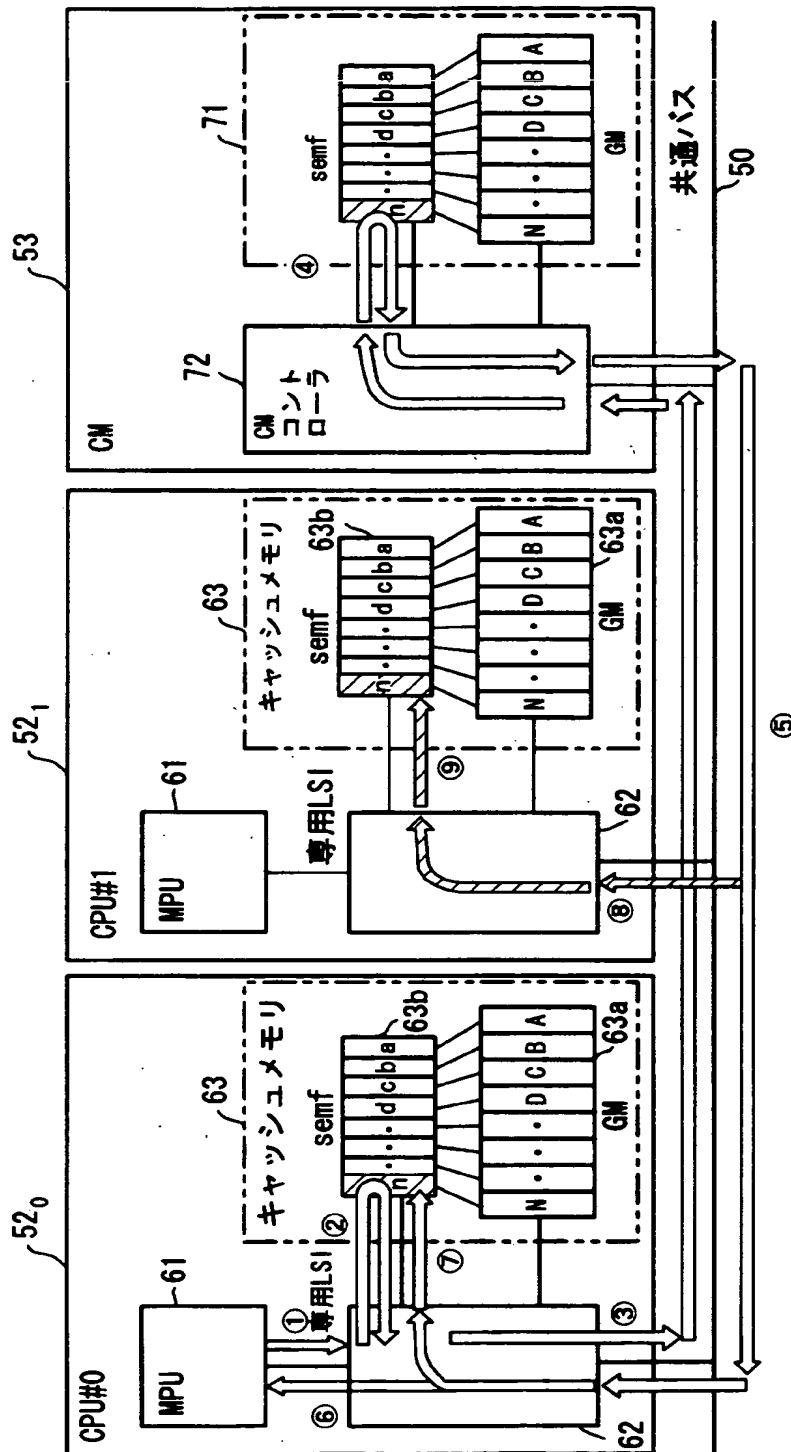
図面

【図 1】

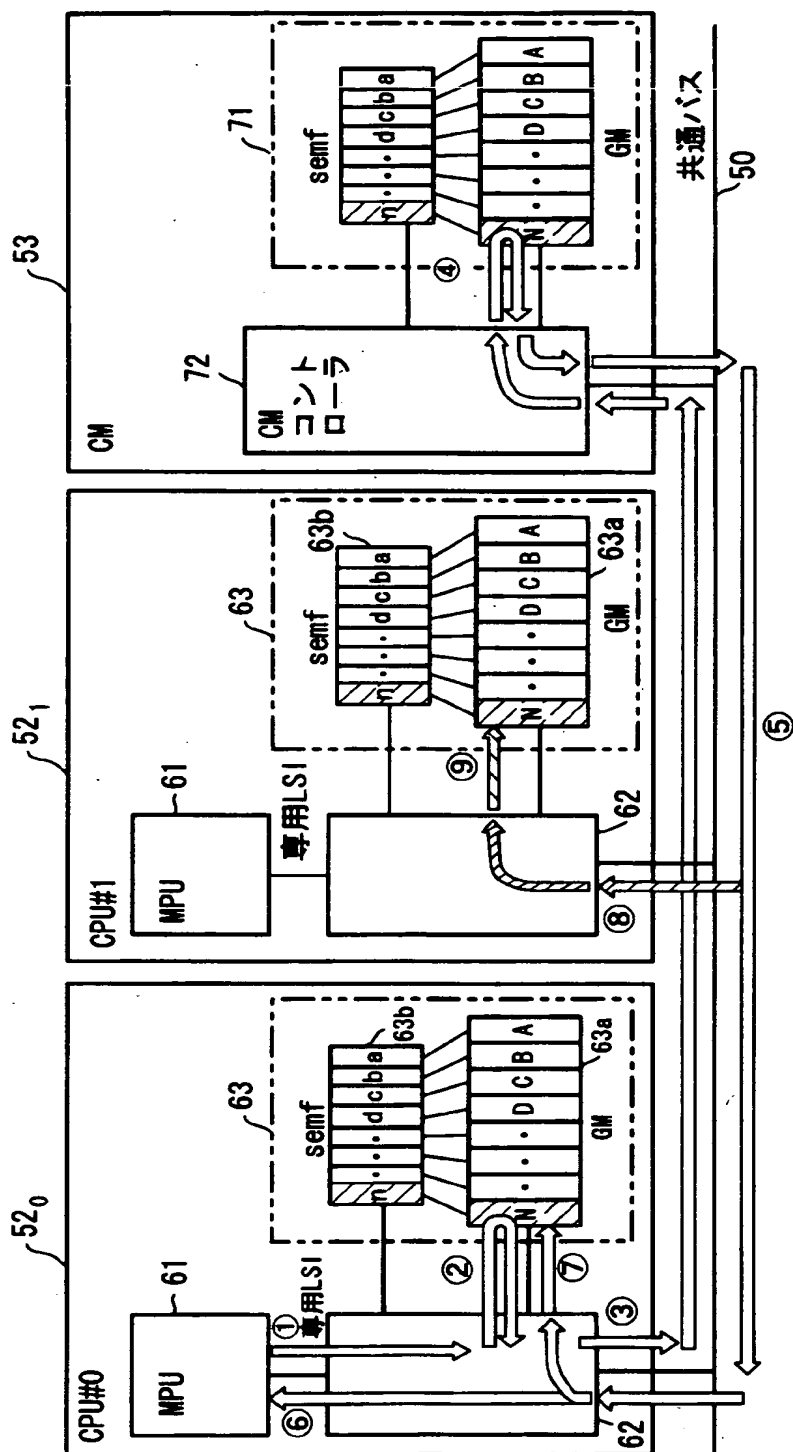
本発明のマルチプロセッサシステムの構成



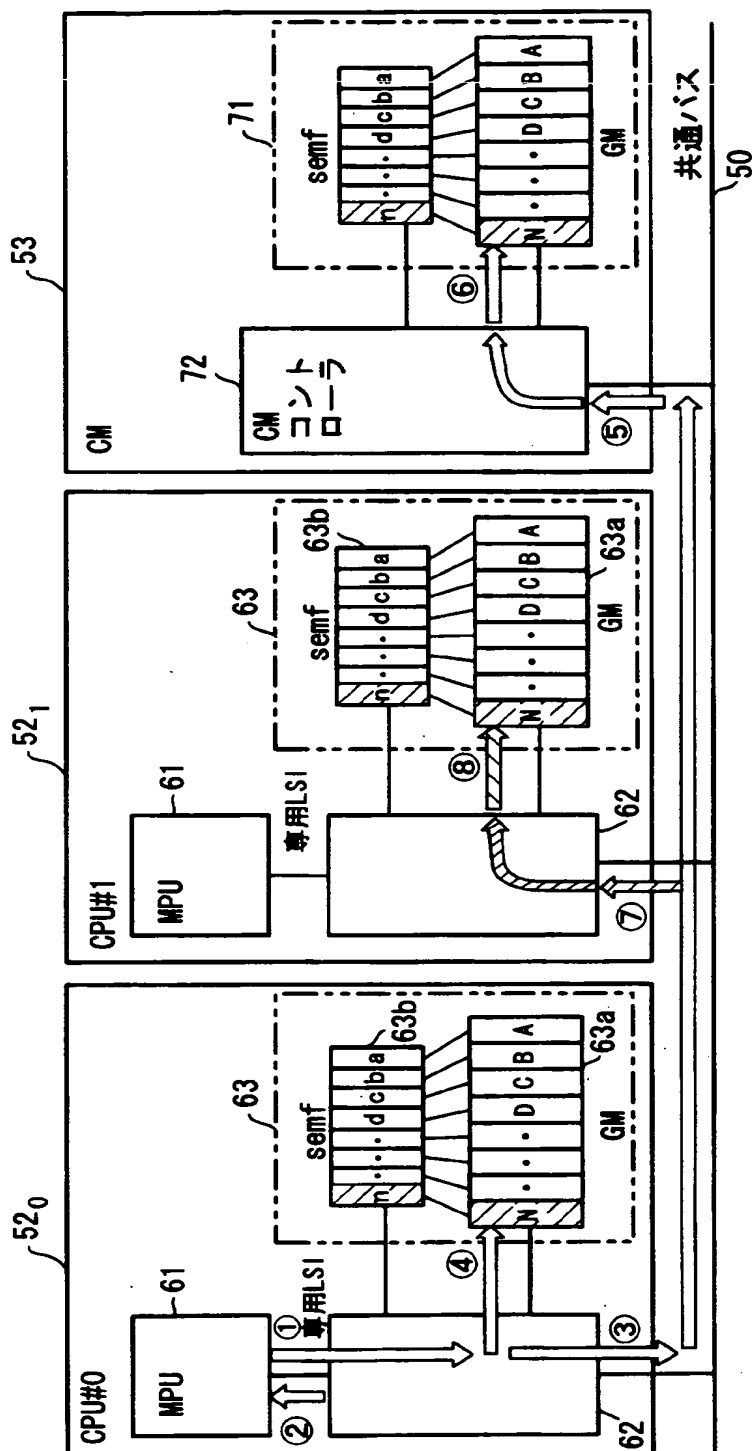
【図 2】



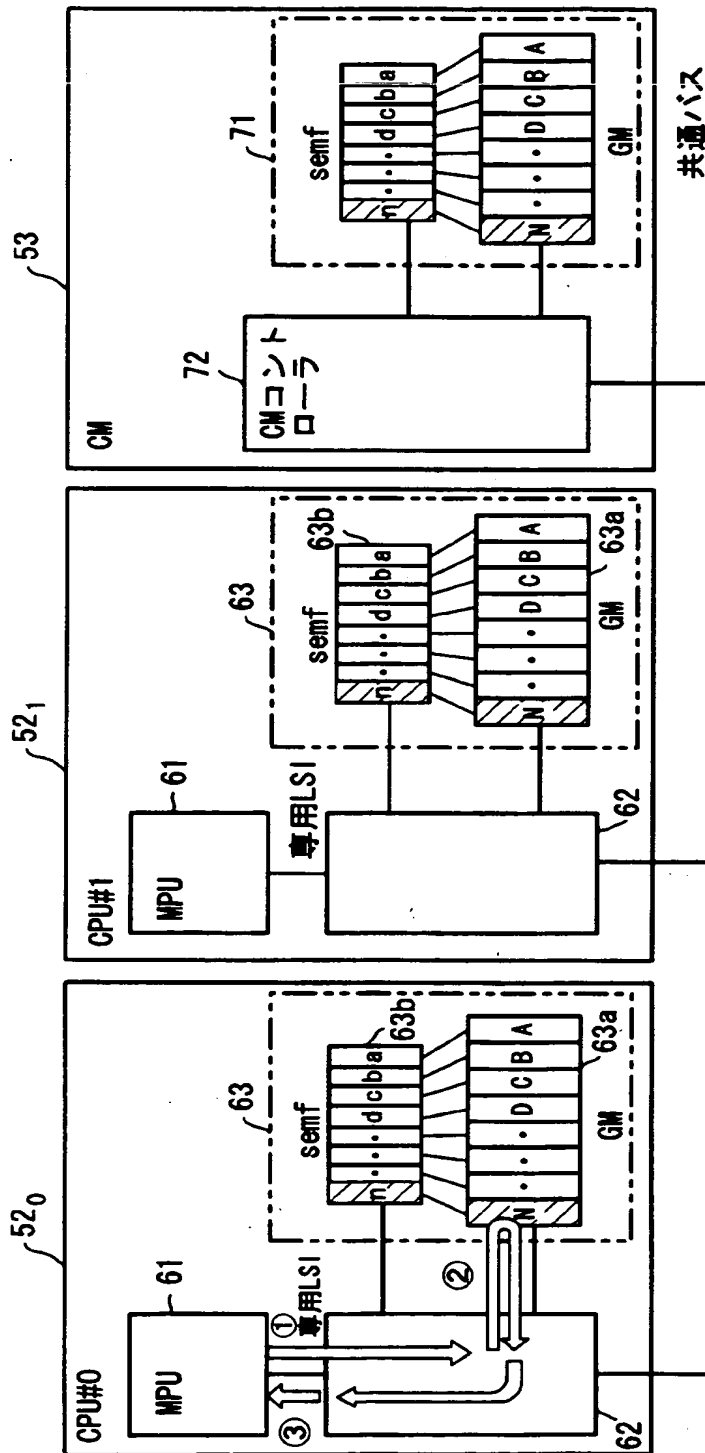
【図 3】



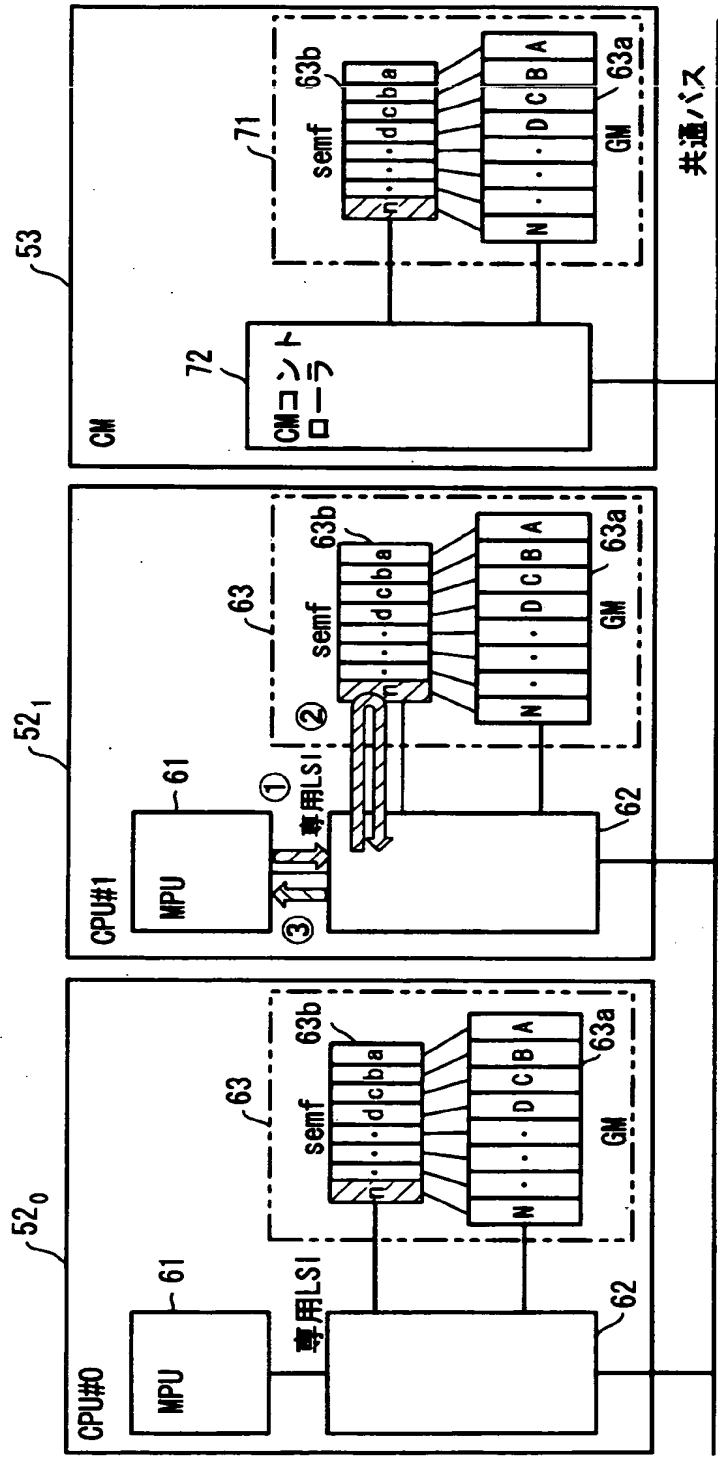
【図 4】



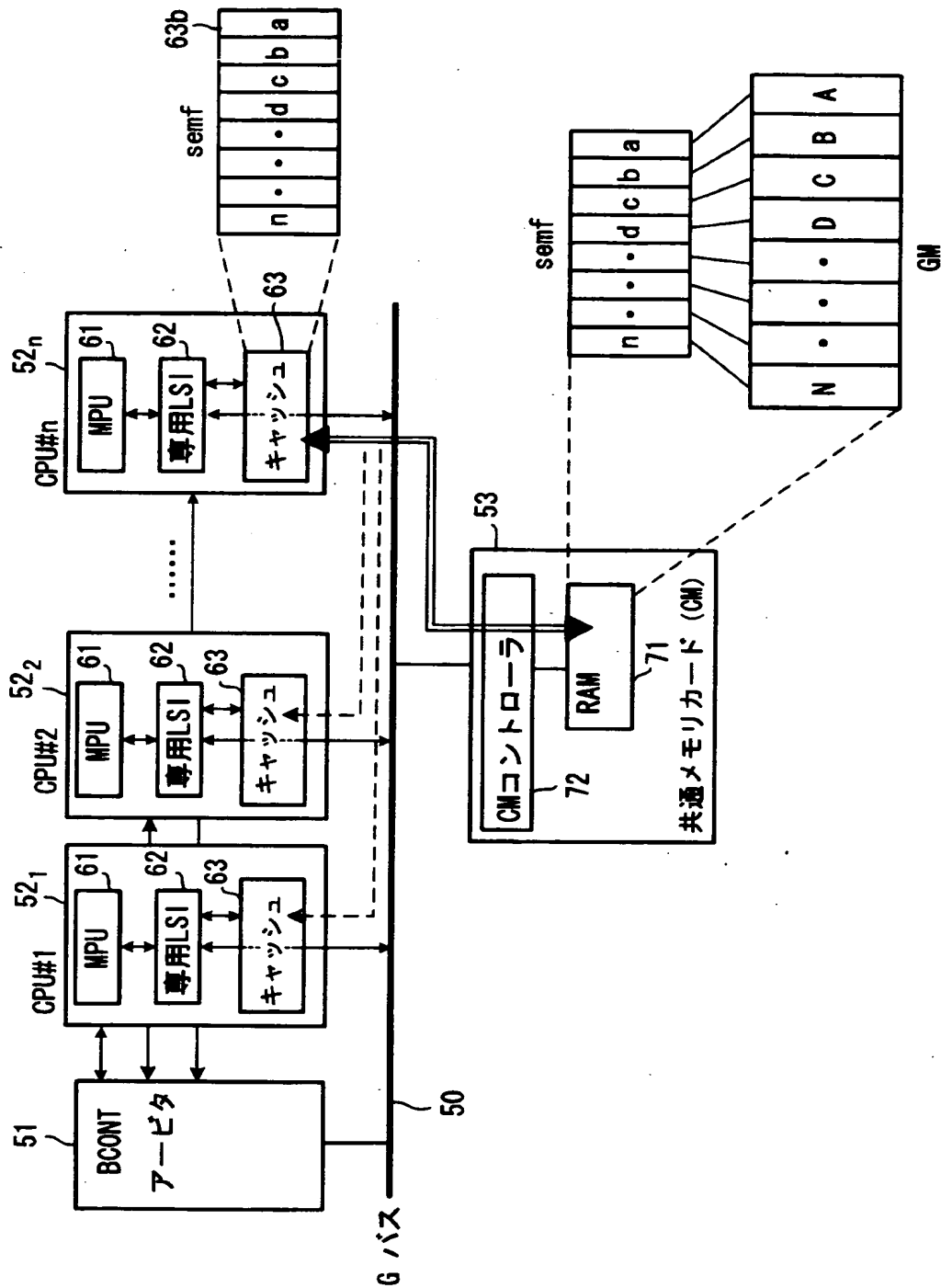
【図 5】



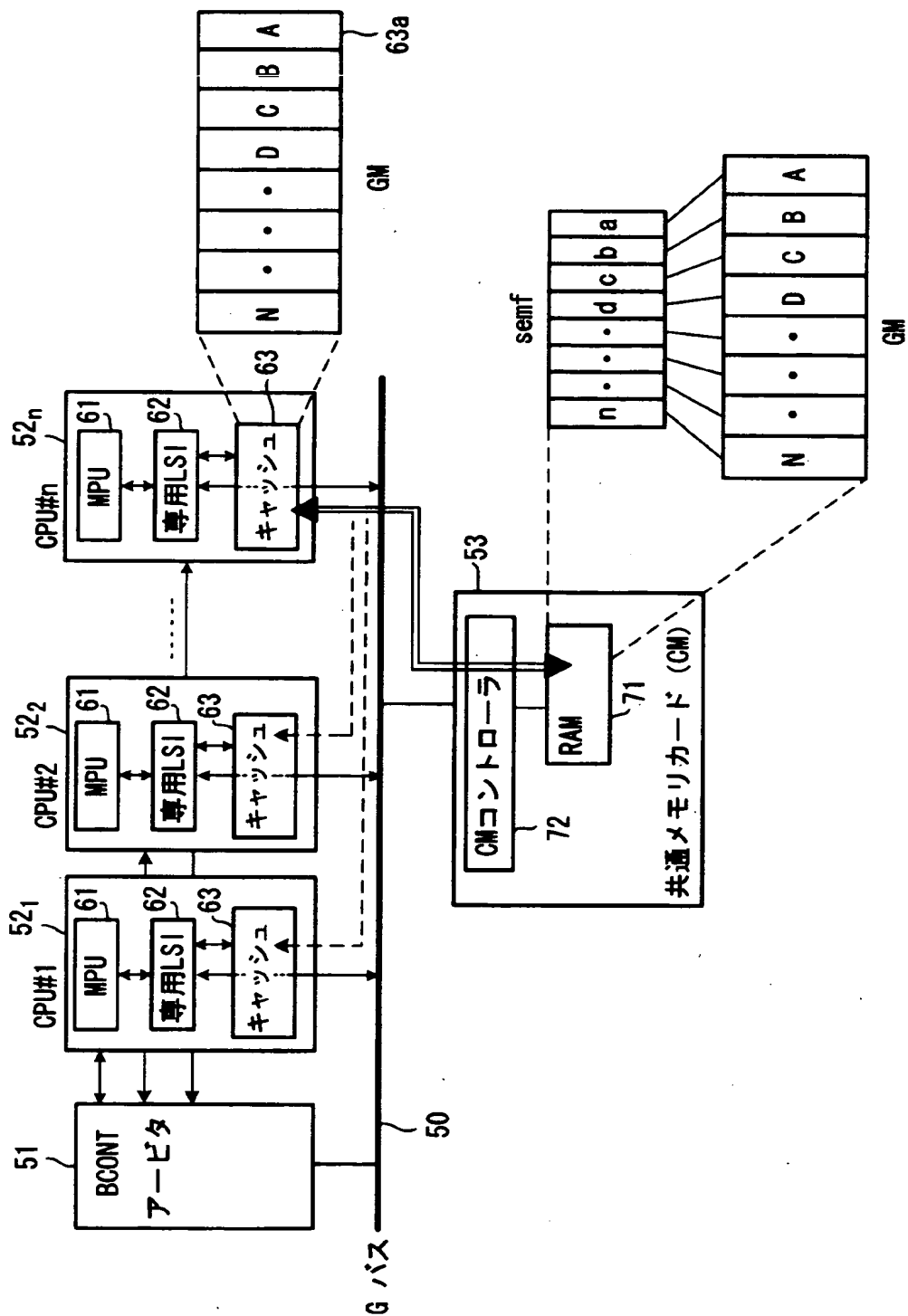
【図 6】



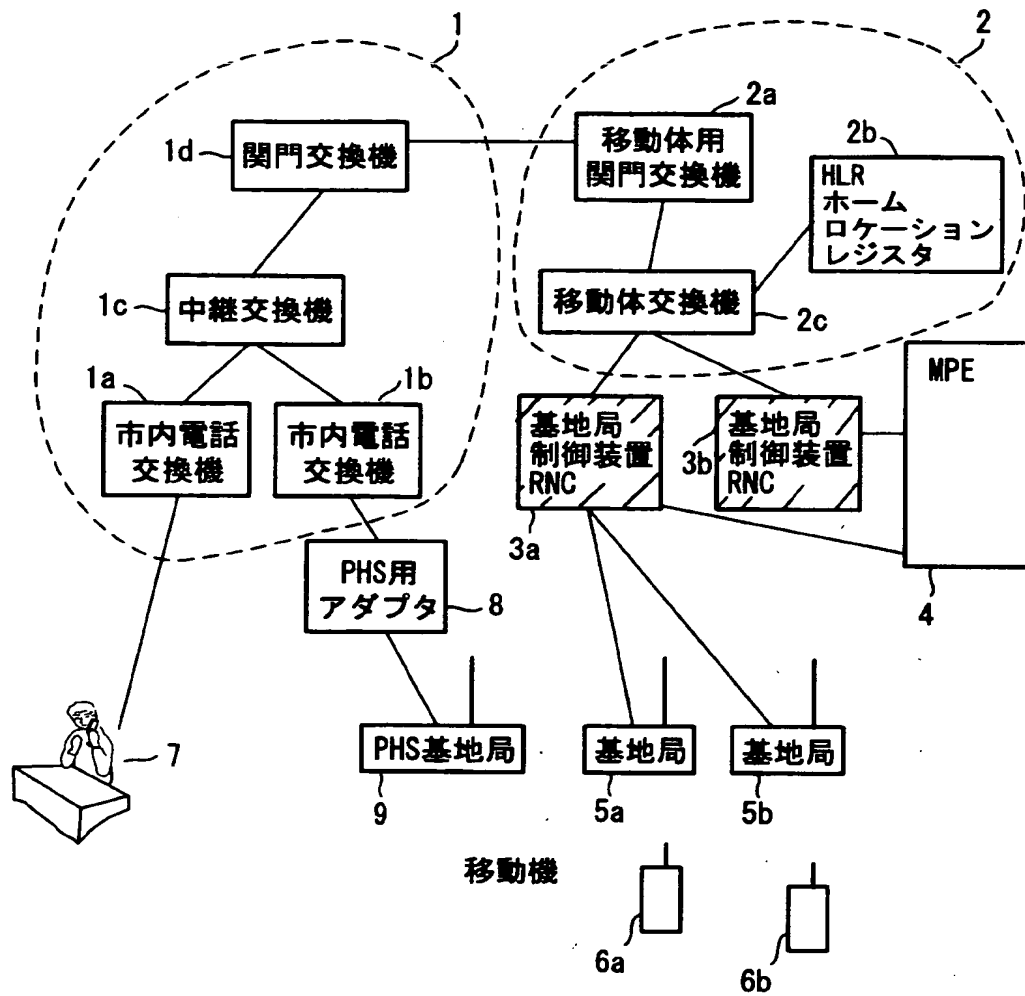
【図 7】



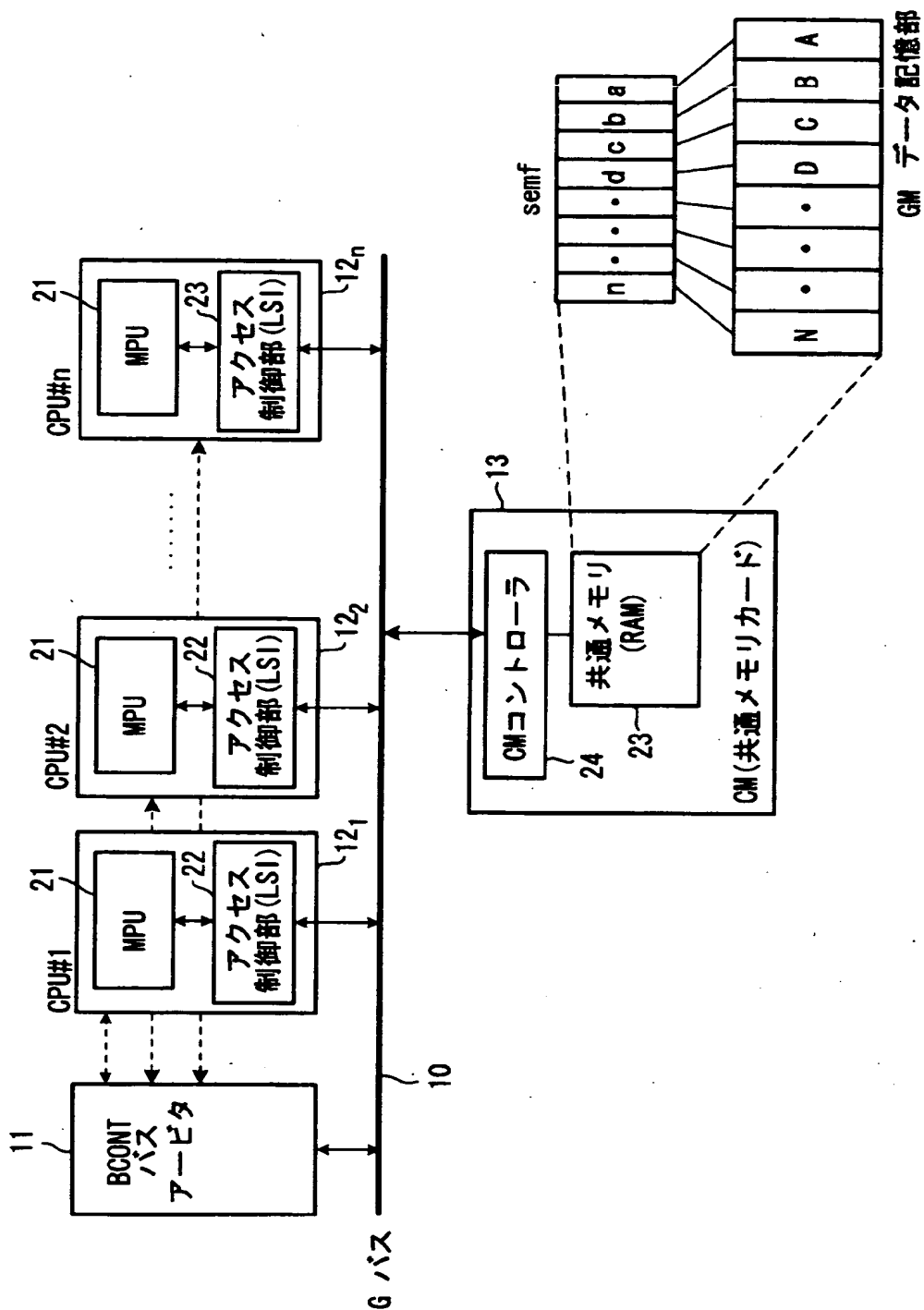
【図 8】



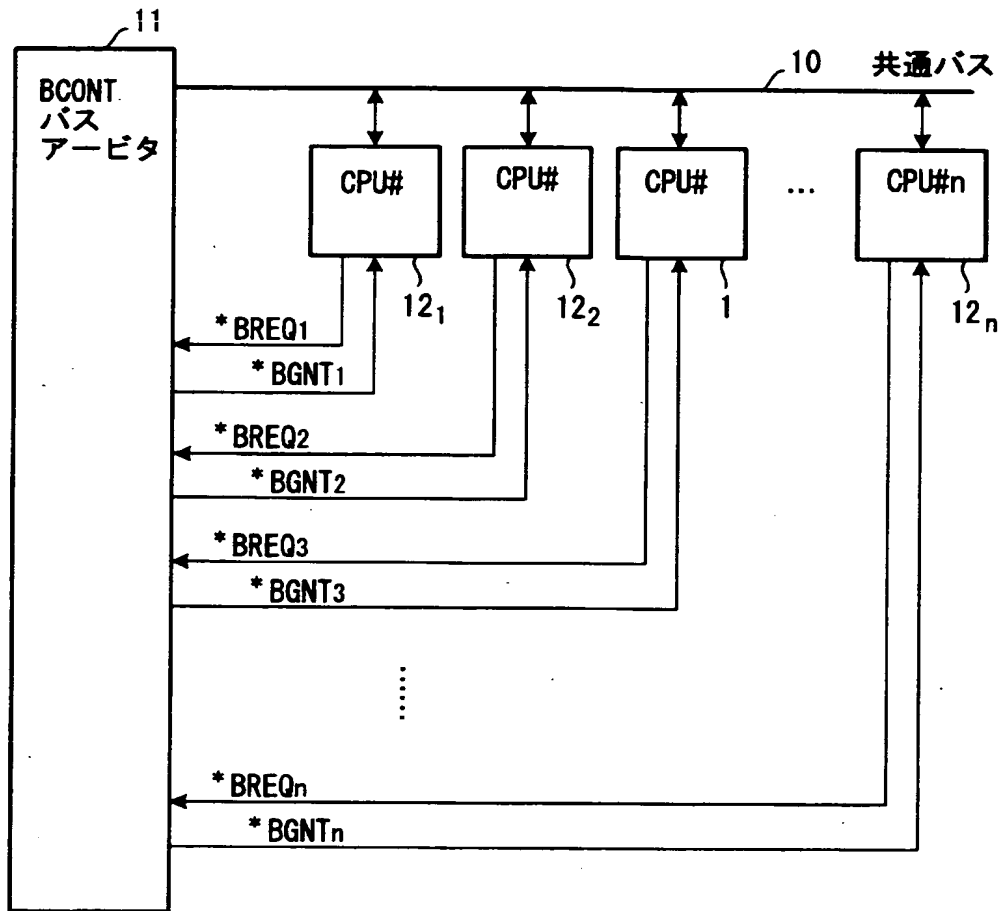
【図 9】



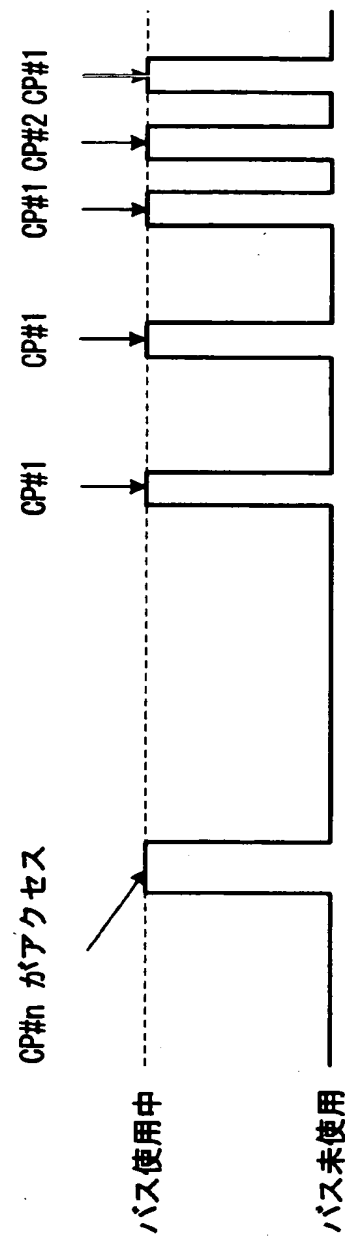
【図 10】



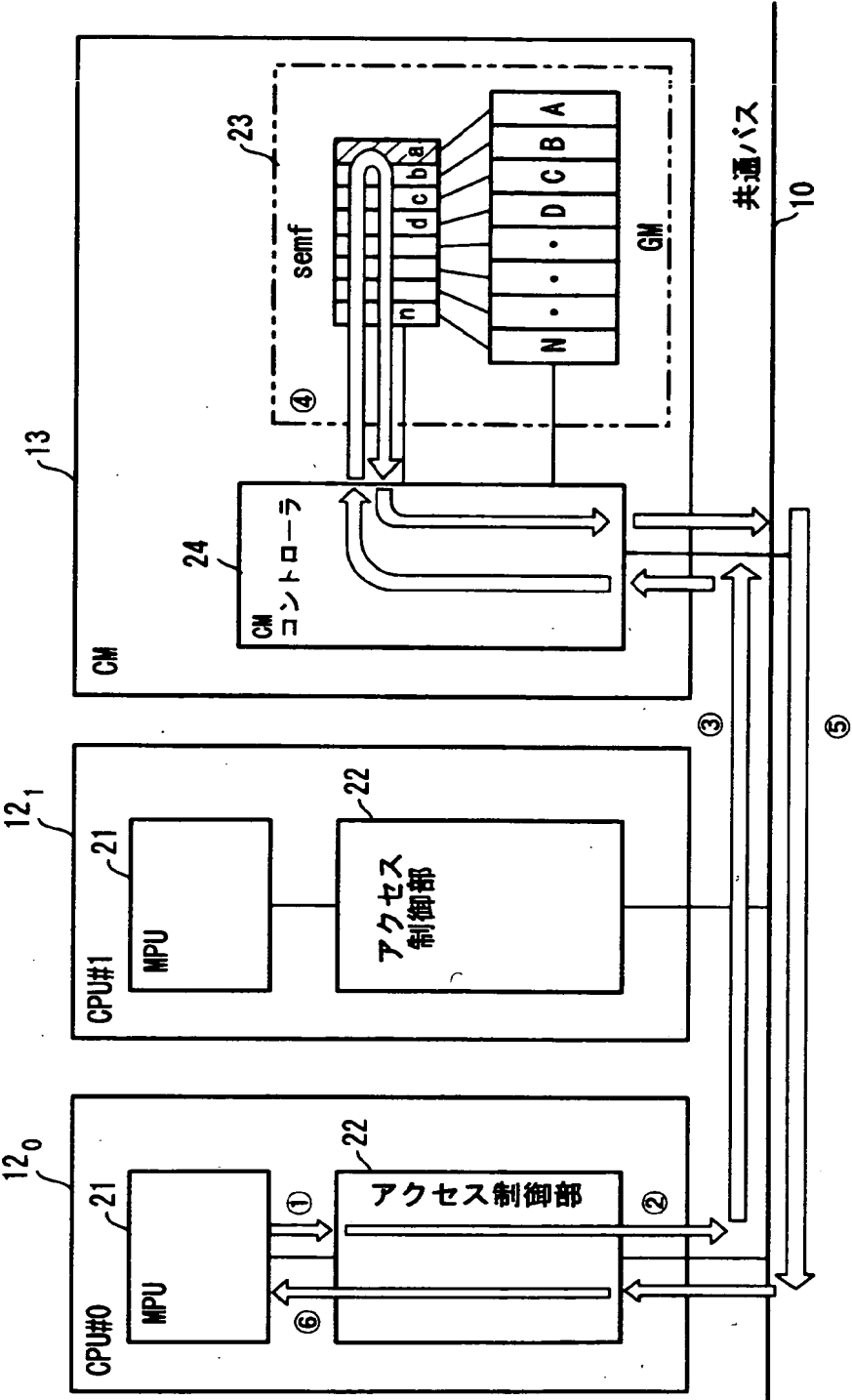
【図 11】



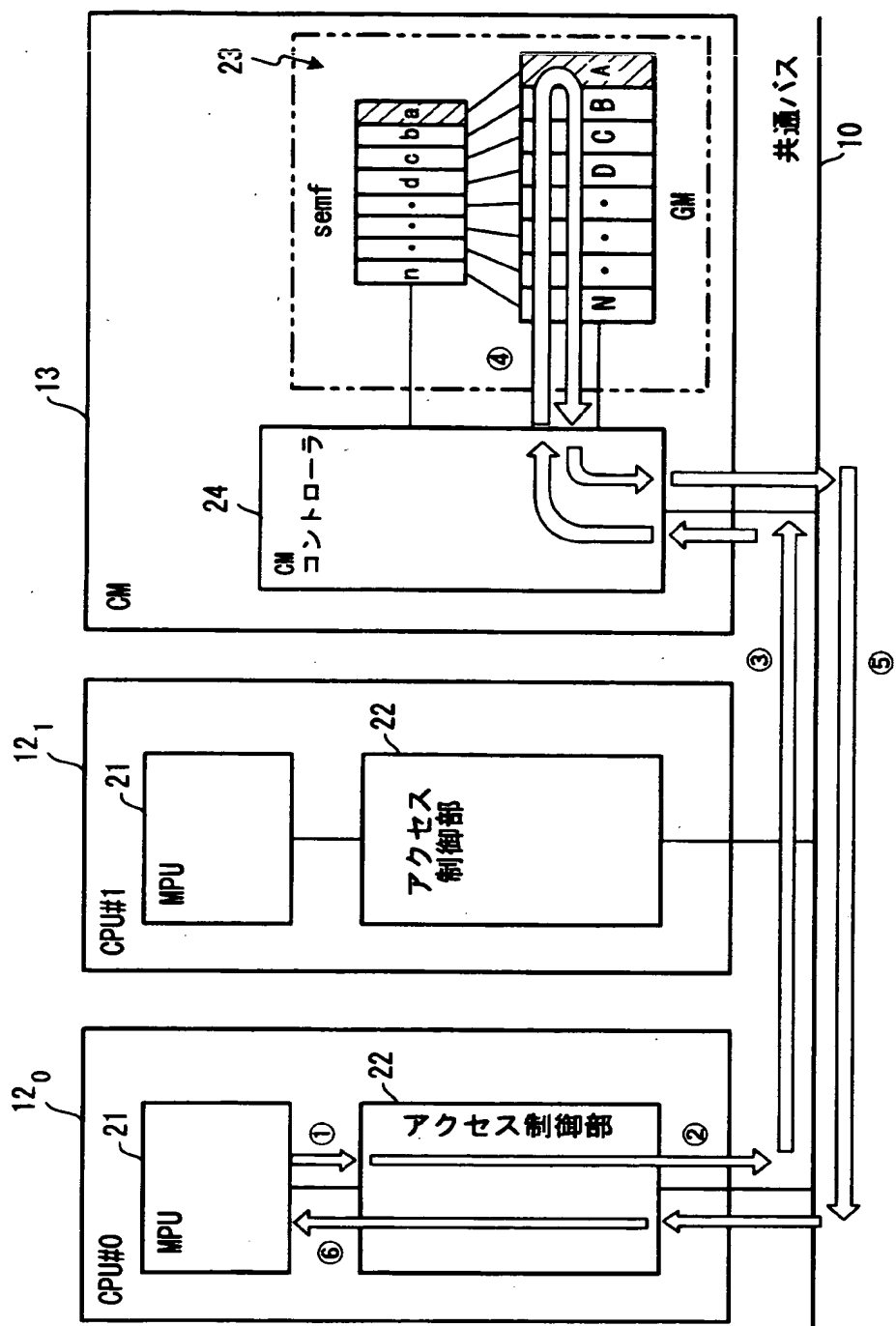
【図 1 2】



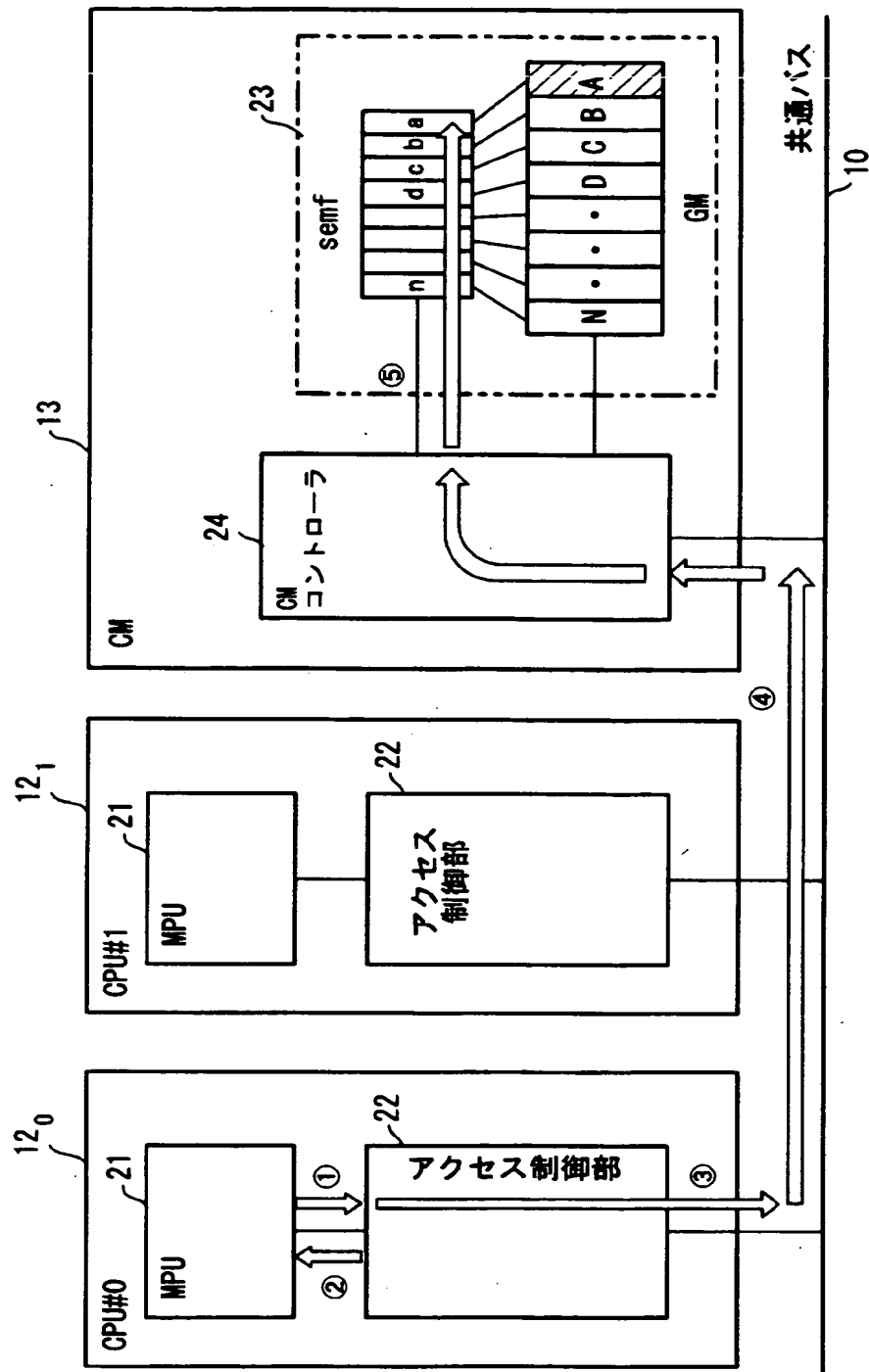
【図 13】



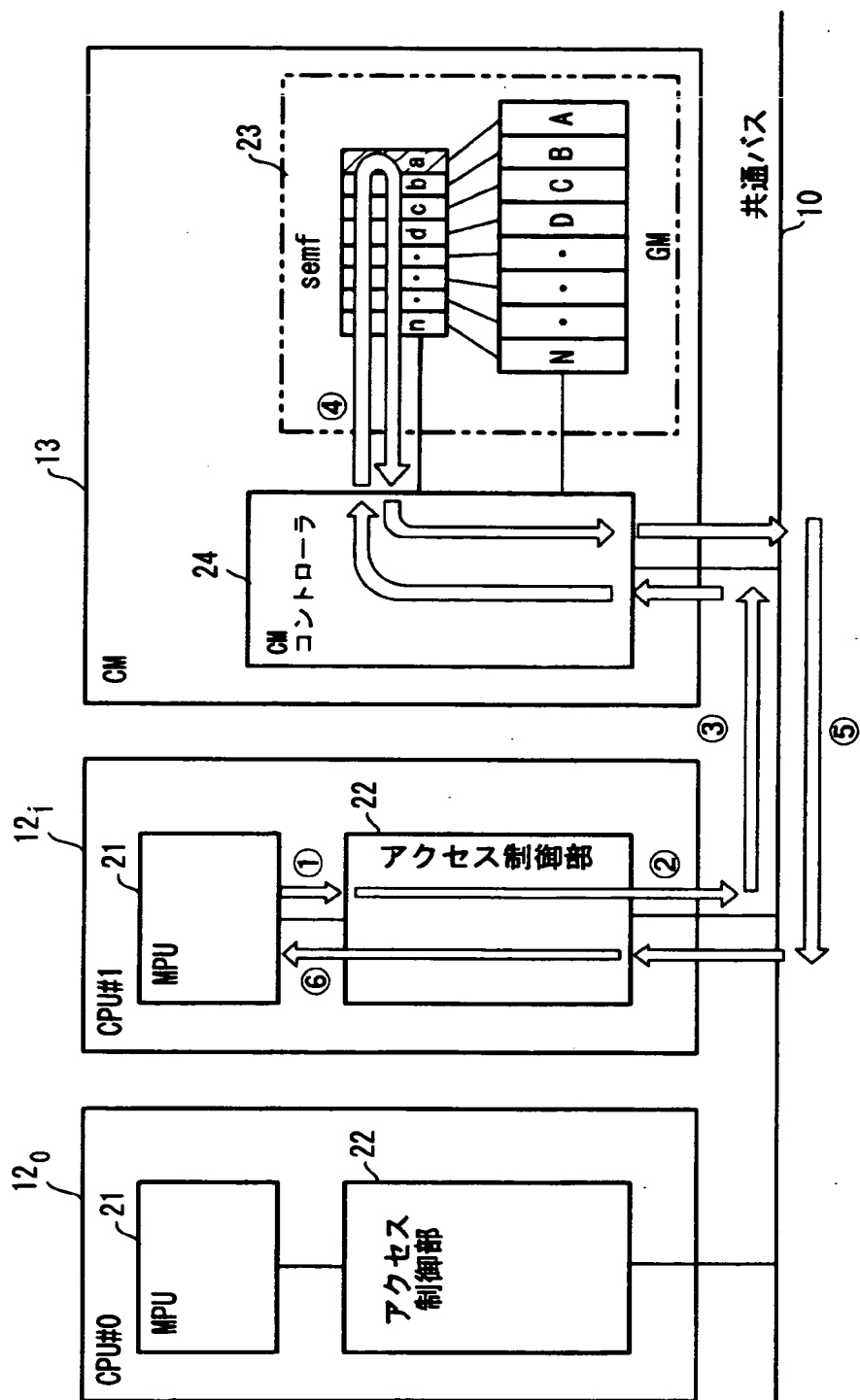
【図 14】



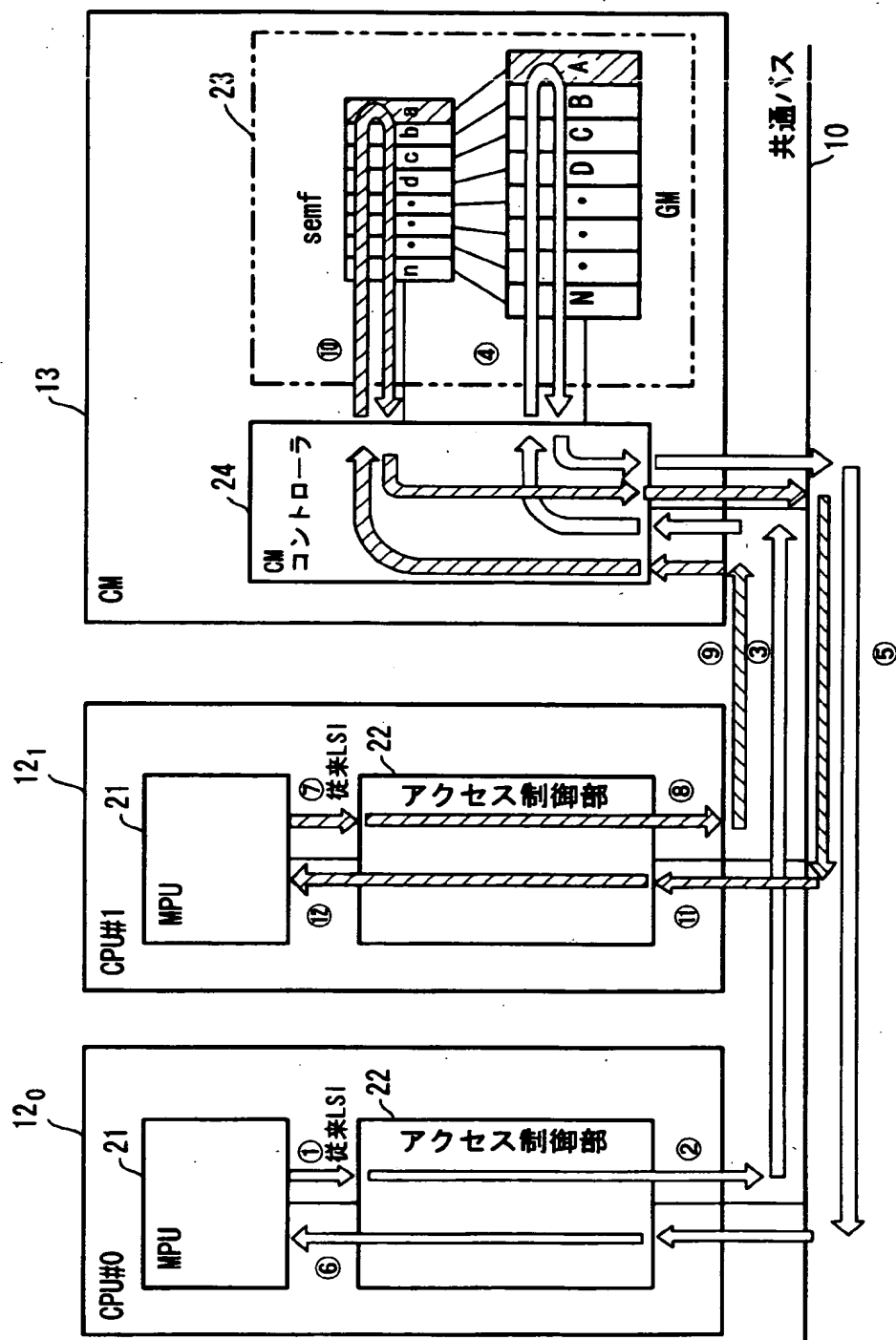
【図 15】



【図 16】



【図 17】



【書類名】 要約書

【要約】

【課題】 各プロセッサからの共通メモリへのアクセスが競合しても、共通メモリへのアクセス回数を削減する。

【解決手段】 マルチプロセッサシステムにおいて、共通メモリにデータを記憶する多数のデータ領域と該データ領域毎の使用中表示する制御情報を記憶する制御情報領域を設け、各プロセッサに共通メモリと同等の記憶ユニットとアクセス制御部を設け、アクセス権のないプロセッサのアクセス制御部は、共通バス上を流れるデータ及びアドレスを監視し、共通メモリへの書き込みデータ及び共通メモリからの読み取りデータを取り込んで自プロセッサ内の記憶ユニットに格納して共通メモリと同一の内容を保持する。

【選択図】 図1

認定・付加情報

特許出願の番号	特願 2002-345371
受付番号	50201801113
書類名	特許願
担当官	末武 実 1912
作成日	平成14年12月10日

<認定情報・付加情報>

【特許出願人】

【識別番号】	000005223
【住所又は居所】	神奈川県川崎市中原区上小田中4丁目1番1号
【氏名又は名称】	富士通株式会社

【代理人】

申請人	
【識別番号】	100084711
【住所又は居所】	千葉県千葉市花見川区幕張本郷1丁目14番10号 幸栄パレス202 齋藤特許事務所
【氏名又は名称】	斉藤 千幹

次頁無

特願 2002-345371

出 願 人 履 歷 情 報

識別番号

[000005223]

1. 変更年月日

1990年 8月24日

[変更理由]

新規登録

住 所

神奈川県川崎市中原区上小田中1015番地

氏 名

富士通株式会社

2. 変更年月日

1996年 3月26日

[変更理由]

住所変更

住 所

神奈川県川崎市中原区上小田中4丁目1番1号

氏 名

富士通株式会社